Computer Science Department

TECHNICAL REPORT

Optimal Preconditioners of a Given Sparsity Pattern

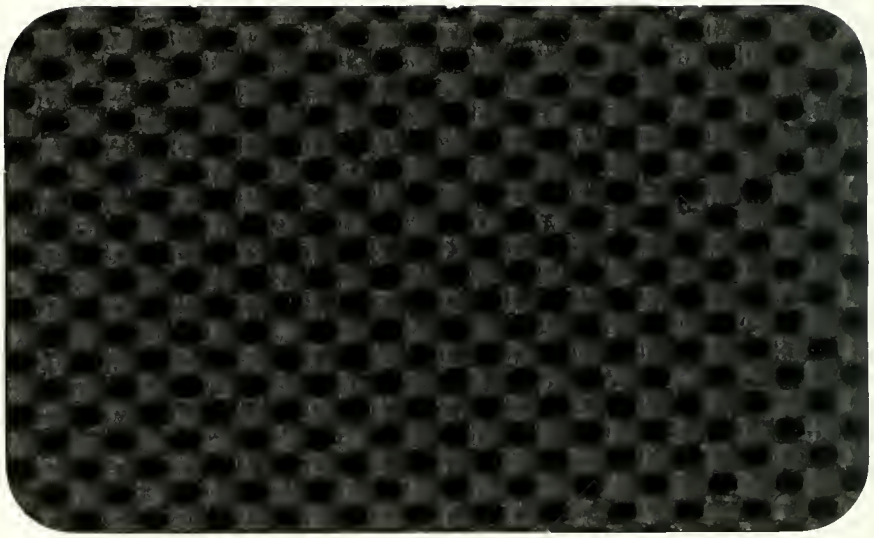
A. Greenbaum
G. Rodrigue

Technical Report 431

February 1989

NEW YORK UNIVERSITY

Department of Computer Science
Courant Institute of Mathematical Sciences
251 MERCER STREET, NEW YORK, N.Y. 10012



Optimal Preconditioners of a Given Sparsity Pattern

A. Greenbaum
G. Rodrigue

Technical Report 431

February 1989

Optimal Preconditioners of a Given Sparsity Pattern

A. Greenbaum †

New York University
Courant Institute of Mathematical Sciences
251 Mercer Street
New York, NY 10012

G. H. Rodrigue

University of California
Lawrence Livermore National Laboratory
P.O. Box 808
Livermore, CA 94550

1. Introduction.

In recent years much research has focused on the problem of finding efficient preconditioners to use with various iterative methods for solving linear systems. Examples of preconditioners, or of iterative methods that can be viewed as using special preconditioners, include the incomplete Cholesky factorization [19], the SSOR preconditioner [25], multigrid methods [2], domain decomposition techniques [1], hierarchical basis functions [26], and many, many more.

An efficient preconditioner M for a matrix A must possess two properties:

1.) Linear systems with coefficient matrix M must be relatively easy to solve; and

2.) The matrix M must "approximate" the matrix A.

Many of the preconditioners that have been proposed are easy to solve because of their sparsity pattern or because they are products of known lower and upper triangular matrices with simple sparsity patterns.

The sense in which M should "approximate" A differs according to the iterative method to be used. For simple iterative refinement methods ($x^k = x^{k-1} + M^{-1}(b - Ax^{k-1})$), the asymptotic convergence rate is determined by the spectral radius

$$\rho(I - M^{-1}A).$$

For fast asymptotic convergence, this quantity should be small.

When the matrices A and M are symmetric and positive definite, this basic iterative method can be accelerated through use of the Chebyshev or conjugate gradient iteration. For the Chebyshev or conjugate gradient methods, the ratio

$$\kappa(M^{-1}A) = \frac{\lambda_{max}(M^{-1}A)}{\lambda_{min}(M^{-1}A)}$$

---

† This work was supported by the Advanced Research Projects Agency of the Department of Defense under contract F49620-87-C-0065 and by the Applied Mathematical Sciences Program of the U.S. Depart-

of the largest to the smallest eigenvalue of $M^{-1}A$ enters into an upper bound for the error

$$\frac{\|e^k\|_A}{\|e^0\|_A} \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k.$$

To make this bound small, κ should be close to 1. This bound is *sharp* for the Chebyshev method, in the sense that there is an initial guess for which the bound will be attained at every step. It is not sharp for the conjugate gradient method. A sharp error bound for the conjugate gradient method is more complicated [13], involving the distribution of all eigenvalues of $M^{-1}A$, but a condition number[1] $\kappa(M^{-1}A)$ close to 1 is *sufficient* to ensure fast convergence of this algorithm as well, even when the effects of finite precision arithmetic are taken into account [14]. Therefore, we will define "optimality" in terms of the condition number $\kappa(M^{-1}A)$ and minimization of this quantity will be our goal.

It is well-known that the largest eigenvalue of a symmetric matrix S is a convex function of the elements of S. Although the function is not differentiable at points where eigenvalues coalesce, (and one can usually expect the minimum to occur at such a point), the problem of minimizing this function can be handled numerically using optimization techniques. If the elements of S are affine functions of a vector x of unknowns

$$S(x) = S_0 + \sum_{k=1}^{m} S_k x_k$$

then the largest eigenvalue of $S(x)$ will be a convex function of x. Given the matrices S_k, $k=0,1,\ldots,m$ an algorithm due to M. Overton [20] can be used to find the vector x for which the spectral radius of $S(x)$ is minimal. The algorithm is asymptotically quadratically convergent and second derivatives are not required to obtain this quadratic convergence rate in many cases. The code uses a variant of Newton's method to minimize a related nonlinear but essentially differentiable function. In this paper, we report results using the Overton optimization code to find optimal preconditioners of a given sparsity pattern.

To see how the preconditioning problem can fit into this framework, we will need a few simple results. Given a symmetric positive definite matrix A and a sparsity pattern for the symmetric preconditioner M, we would like to find the matrix M of the given form which minimizes $\rho(I-M^{-1}A)$ or $\kappa(M^{-1}A)$, as explained above. The matrix $I-M^{-1}A$ is not symmetric, and its elements are not affine functions of the elements of M, but the following theorem relates the minimization of $\rho(I-M^{-1}A)$ to the minimization of the largest eigenvalue of $I - L^{-1}ML^{-T}$, where LL^T is a factorization (e.g., the Cholesky factorization) of A. We start with the following simple lemma.

Lemma 1. For a given symmetric matrix Q with nonnegative eigenvalues, the constant c which minimizes $\rho(I-cQ)$ is

$$c = \frac{2}{\lambda_{max}(Q) + \lambda_{min}(Q)},$$

where $\lambda_{min}(Q)$ is the smallest and $\lambda_{max}(Q)$ the largest eigenvalue of Q.

---

ment of Energy under contract DE-AC02-76ER03077.

[1] When referring to the condition number $\kappa(M^{-1}A)$, we actually mean the ratio of largest to smallest eigenvalue of $M^{-1}A$, or, the condition number in the 2-norm of $M^{-1/2}AM^{-1/2}$. Since the matrices we consider are all symmetric and positive definite, this should cause no confusion.

Proof: For c as defined above, the spectral radius of $I-cQ$ is given by

$$\rho(I-cQ) = 1 - c\lambda_{\min}(Q) = c\lambda_{\max}(Q) - 1.$$

For any other constant $c' \neq c$, one of the following two inequalities must hold:

$$c'<c \;\rightarrow\; 1 - c'\lambda_{\min}(Q) > 1 - c\lambda_{\min}(Q), \quad \textit{or} \quad c'>c \;\rightarrow\; c'\lambda_{\max}(Q) - 1 > c\lambda_{\max}(Q) - 1,$$

and so the spectral radius of $I-c'Q$ must satisfy

$$\rho(I-c'Q) > \rho(I-cQ).$$

□

Theorem 1. Let $A=LL^T$ be a factorization of the symmetric positive definite matrix A, and let M_1 and M_2 be two symmetric positive semi-definite matrices satisfying

$$\rho(I-c_1L^{-1}M_1L^{-T}) < \rho(I-c_2L^{-1}M_2L^{-T}),$$

where the constants c_1 and c_2, minimize these spectral radii, as described in Lemma 1; i.e.,

$$c_i = \frac{2}{\lambda_{\max}(L^{-1}M_iL^{-T}) + \lambda_{\min}(L^{-1}M_iL^{-T})}, \quad i=1,2.$$

Then M_1 and M_2 also satisfy

$$\rho(I-\hat{c}_1M_1^{-1}A) < \rho(I-\hat{c}_2M_2^{-1}A),$$

where $\hat{c}_1$ and $\hat{c}_2$ minimize these spectral radii; i.e.,

$$\hat{c}_i = \frac{2}{\lambda_{\max}(M_i^{-1}A) + \lambda_{\min}(M_i^{-1}A)}, \quad i=1,2.$$

Proof: Because of the choice of c_1 and c_2, we have

$$\rho(I-c_iL^{-1}M_iL^{-T}) = \frac{\lambda_{\max}(L^{-1}M_iL^{-T}) - \lambda_{\min}(L^{-1}M_iL^{-T})}{\lambda_{\max}(L^{-1}M_iL^{-T}) + \lambda_{\min}(L^{-1}M_iL^{-T})}, \quad i=1,2.$$

The eigenvalues of $L^{-1}M_iL^{-T}$, i=1,2, are just the inverses of the eigenvalues of $L^TM_i^{-1}L$, or, of $M_i^{-1}A$. Hence we have

$$\rho(I-c_iL^{-1}M_iL^{-T}) = \frac{\lambda_{\max}(M_i^{-1}A) - \lambda_{\min}(M_i^{-1}A)}{\lambda_{\max}(M_i^{-1}A) + \lambda_{\min}(M_i^{-1}A)},$$

and because of the choice of $\hat{c}_1$ and $\hat{c}_2$, the right-hand side is equal to the spectral radius of

$I-\hat{c}_i M_i^{-1}A$, i=1,2. Thus, we have

$$\rho(I-\hat{c}_1 M_1^{-1}A) = \rho(I-c_1 L^{-1}M_1 L^{-T}) < \rho(I-c_2 L^{-1}M_2 L^{-T}) = \rho(I-\hat{c}_2 M_2^{-1}A). \qquad \square$$

The symmetric matrix $I-L^{-1}ML^{-T}$ is an affine function of the elements of M. Theorem 1 shows that if M_1 minimizes the spectral radius of this matrix over all symmetric preconditioners M belonging to some set which also contains all constant multiples of its members, then there is a constant $\hat{c}_1$ such that $\hat{c}_1 M_1$ minimizes the spectral radius of $I-M^{-1}A$ over this set. The constant $\hat{c}_1$ is defined in Lemma 1. The following theorem shows that this same matrix M_1 also minimizes the ratio of largest to smallest eigenvalue, $\kappa(M^{-1}A)$.

Theorem 2. Let $A=LL^T$ be a factorization of the symmetric positive definite matrix A, and let M_1 and M_2 be two symmetric positive semi-definite matrices satisfying

$$\rho(I-L^{-1}M_1 L^{-T}) < \rho(I-c_2 L^{-1}M_2 L^{-T}),$$

where the constant c_2 minimizes this spectral radius; i.e.,

$$c_2 = \frac{2}{\lambda_{max}(L^{-1}M_2 L^{-T}) + \lambda_{min}(L^{-1}M_2 L^{-T})}.$$

Then M_1 and M_2 also satisfy

$$\kappa(M_1^{-1}A) < \kappa(M_2^{-1}A).$$

Proof: By the choice of c_2 we have

$$\rho(I-c_2 L^{-1}M_2 L^{-T}) = 1 - \frac{2\lambda_{min}(L^{-1}M_2 L^{-T})}{\lambda_{max}(L^{-1}M_2 L^{-T}) + \lambda_{min}(L^{-1}M_2 L^{-T})} = \frac{2\lambda_{max}(L^{-1}M_2 L^{-T})}{\lambda_{max}(L^{-1}M_2 L^{-T}) + \lambda_{min}(L^{-1}M_2 L^{-T})} - 1 .$$

The hypothesis then implies that the eigenvalues of $L^{-1}M_1 L^{-T}$ obey

$$1 - \lambda_{min}(L^{-1}M_1 L^{-T}) < 1 - \frac{2\lambda_{min}(L^{-1}M_2 L^{-T})}{\lambda_{max}(L^{-1}M_2 L^{-T}) + \lambda_{min}(L^{-1}M_2 L^{-T})},$$

$$\lambda_{max}(L^{-1}M_1 L^{-T}) - 1 < \frac{2\lambda_{max}(L^{-1}M_2 L^{-T})}{\lambda_{max}(L^{-1}M_2 L^{-T}) + \lambda_{min}(L^{-1}M_2 L^{-T})} - 1 ,$$

or,

$$\lambda_{min}(L^{-1}M_1 L^{-T}) > \frac{2\lambda_{min}(L^{-1}M_2 L^{-T})}{\lambda_{max}(L^{-1}M_2 L^{-T}) + \lambda_{min}(L^{-1}M_2 L^{-T})},$$

$$\lambda_{max}(L^{-1}M_1 L^{-T}) < \frac{2\lambda_{max}(L^{-1}M_2 L^{-T})}{\lambda_{max}(L^{-1}M_2 L^{-T}) + \lambda_{min}(L^{-1}M_2 L^{-T})}.$$

Dividing the second inequality by the first then gives

$$\frac{\lambda_{max}(L^{-1}M_1L^{-T})}{\lambda_{min}(L^{-1}M_1L^{-T})} < \frac{\lambda_{max}(L^{-1}M_2L^{-T})}{\lambda_{min}(L^{-1}M_2L^{-T})},$$

and since the eigenvalues of $L^{-1}M_iL^{-T}$, i=1,2, are just the inverses of the eigenvalues of $M_i^{-1}A$, the desired result follows. □

In the following examples we find the matrix M in a given set which minimizes $\rho(I-L^{-1}ML^{-T})$. According to Theorem 2, this also minimizes $\kappa(M^{-1}A)$ over the set. Theorems 1 and 2, together with the previous discussion, justify referring to this matrix (or an appropriate constant multiple of this matrix) as the "optimal" preconditioner from the set for use with the Chebyshev method or with iterative refinement. Other criteria are possible for defining a good preconditioner for the conjugate gradient method [6,21], but this is a reasonable and simple measure, and if there is a matrix M in the set for which this measure is small, then the conjugate gradient method with this preconditioner will have guaranteed fast convergence.

This same idea of minimizing the condition number was actually pursued for a certain class of preconditioners much earlier by Concus and Golub [5]. They considered 1-D model problems and used a code by Fletcher [10] to find optimal diagonal scalings of the Laplacian to use as a preconditioner.

2. Theoretical Results.

There are few theoretical results concerning optimal preconditioners for most possible sparsity patterns. An exception is the case of diagonal and block diagonal preconditioners. Van der Sluis [22] proved the following theorem about diagonal scaling of a symmetric positive definite matrix A:

Theorem (Van der Sluis). Let D be the diagonal of the symmetric positive definite matrix A, and let $\hat{D}$ be any other positive definite diagonal matrix. Then $\kappa(D^{-1/2}AD^{-1/2})$ satisfies

$$\kappa(D^{-1/2}AD^{-1/2}) \leq m\ \kappa(\hat{D}A\hat{D}),$$

where m is the maximum number of nonzeros in any row of A.

Thus, $D=diag(A)$ approximately minimizes $\kappa(M^{-1}A)$ over all *diagonal* preconditioners M. When the matrix A also possesses property-A, a stronger result holds [11]:

Theorem (Forsythe & Strauss). Using the above notation, if the symmetric positive definite matrix A has property-A, then $\kappa(D^{-1/2}AD^{-1/2})$ satisfies

$$\kappa(D^{-1/2}AD^{-1/2}) \leq \kappa(\hat{D}A\hat{D}).$$

In this case, then, $D = diag(A)$ is the *optimal* diagonal preconditioner for the matrix A.

A generalization of the Van der Sluis theorem has also been proved for *block* diagonal preconditioners [7].

Theorem (Demmel). Let D be the block diagonal of the symmetric positive definite matrix A, and let $\hat{D}$ be any other symmetric positive definite block diagonal matrix with the same size blocks. Then $\kappa(D^{-1/2}AD^{-1/2})$ satisfies

$$\kappa(D^{-1/2}AD^{-1/2}) \leq b\ \kappa(\hat{D}A\hat{D}),$$

where b is the number of blocks in D.

A result similar to that of Forsythe and Strauss has also been proved for block diagonal preconditioners [9], when the matrix A is *block* 2-cyclic and is permuted into the form

$$A = \begin{bmatrix} D_1 & C^T \\ C & D_2 \end{bmatrix} \qquad D_i = \begin{pmatrix} D_{i,1} & 0 & \cdots & 0 \\ 0 & D_{i,2} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & D_{i,r_i} \end{pmatrix} \quad i=1,2. \tag{2.1}$$

Theorem (Eisenstat,Lewis,Schultz). Let A be of the form (2.1), and let D be the block diagonal matrix whose diagonal blocks are $\{D_{1,1}, \cdots, D_{1,r_1}, D_{2,1}, \ldots, D_{2,r_2}\}$. Let $\hat{D}$ be any other block diagonal matrix with the same size blocks. Then $\kappa(D^{-1/2}AD^{-1/2})$ satisfies

$$\kappa(D^{-1/2}AD^{-1/2}) \leq \kappa(\hat{D}A\hat{D}).$$

There appears to be little known theoretically about optimal preconditioners of more general sparsity patterns; e.g., tridiagonal or banded. There is, however, a useful theorem due to Varga [23] for comparing "regular splittings", which in some cases, enables one to determine the optimal preconditioner among all regular splittings of a given sparsity pattern.

Definition. For n by n real matrices A, M, and N, $A = M - N$ is a *regular splitting* of the matrix A if M is nonsingular with $M^{-1} \geq 0$, and $N \geq 0$.

Theorem (Varga). Let $A = M_1 - N_1 = M_2 - N_2$ be two regular splittings of A, where $A^{-1} > 0$. If $N_2 \geq N_1 \geq 0$, (and neither N_1 nor $N_2 - N_1$ is the null matrix), then

$$1 > \rho(M_2^{-1}N_2) > \rho(M_1^{-1}N_1) > 0.$$

This theorem implies that among all regular splitting matrices M of a given bandwidth, for example, the optimal one for minimizing the spectral radius of $I - M^{-1}A$ is $M = band(A)$. This matrix is closer element-wise to A than is any other member of the class, and so, by the

theorem, is a better splitting, or, preconditioner. As will be shown in the following examples, however, there may be better banded preconditioners outside the class of regular splittings. This theorem is quite general and important to remember. Unfortunately, however, many of the most effective preconditioners are not regular splittings, and so it is of limited applicability.

It should also be noted that because the set of regular splittings does not contain all constant multiples of its members, the matrix, say, M, from some class of regular splittings, which minimizes $\rho(I-M^{-1}A)$ over that class does *not* necessarily minimize $\kappa(M^{-1}A)$. The hypotheses of Varga's theorem, together with the assumption that A, M_1, and M_2 are symmetric and positive definite, do *not* imply $\kappa(M_1^{-1}A) \leq \kappa(M_2^{-1}A)$. Only if the matrices M_1 and c_2M_2 -- where c_2 minimizes $\rho(I-cM_2^{-1}A)$, as explained in Lemma 1 -- only if these matrices satisfy the hypotheses of Varga's theorem would it also follow, from Theorem 2, that $\kappa(M_1^{-1}A) \leq \kappa(M_2^{-1}A)$. With a weaker assumption about the matrix M_2, however, it can be shown that $\kappa(M_1^{-1}A) < 2\,\kappa(M_2^{-1}A)$.

Theorem 3. Let A, M_1, and M_2 be symmetric positive definite matrices satisfying the hypotheses of Varga's theorem, and suppose the largest eigenvalue of $M_2^{-1}A$ is greater than or equal to 1. (This would be the case, for example, if A and M_2 have at least one diagonal element equal, since then the symmetric matrix N_2 would have a zero diagonal element and hence $M_2^{-1}N_2$ would have a zero eigenvalue.) Then the ratios of largest to smallest eigenvalues of $M_1^{-1}A$ and $M_2^{-1}A$ satisfy

$$\kappa(M_1^{-1}A) < 2\,\kappa(M_2^{-1}A).$$

Proof: Since the elements of $M_2^{-1}N_2$ are nonnegative, the Perron-Frobenius theorem states that its spectral radius is equal to its (algebraically) largest eigenvalue:

$$\rho(M_2^{-1}N_2) = \rho(I-M_2^{-1}A) = 1 - \lambda_{\min}(M_2^{-1}A).$$

The result $\rho(M_1^{-1}N_1) < \rho(M_2^{-1}N_2)$ from Varga's theorem implies

$$1 - \lambda_{\min}(M_1^{-1}A) < 1 - \lambda_{\min}(M_2^{-1}A) \quad \text{and} \quad \lambda_{\max}(M_1^{-1}A) - 1 < 1 - \lambda_{\min}(M_2^{-1}A),$$

or, equivalently,

$$\lambda_{\min}(M_1^{-1}A) > \lambda_{\min}(M_2^{-1}A) \quad \text{and} \quad \lambda_{\max}(M_1^{-1}A) < 2 - \lambda_{\min}(M_2^{-1}A).$$

Dividing the second inequality by the first gives

$$\kappa(M_1^{-1}A) < \kappa(M_2^{-1}A)\cdot \frac{2 - \lambda_{\min}(M_2^{-1}A)}{\lambda_{\max}(M_2^{-1}A)} < \kappa(M_2^{-1}A)\cdot 2,$$

with the last inequality holding because $\lambda_{\max}(M_2^{-1}A) \geq 1$, and $\lambda_{\min}(M_2^{-1}A) > 0$ since $\rho(M_2^{-1}N_2) < 1$. □

A somewhat more general result can be proved for symmetric matrices.

Definition. A splitting $A = M - N$ is said to be a *weak regular splitting* of the matrix A if M is nonsingular with $M^{-1}N \geq 0$.

We then get the following comparison theorem.

Theorem 4. Suppose A and M are symmetric positive definte matrices such that $A = M - N$ is a weak regular splitting of A. Let $M_Q = M + Q$ for some symmetric matrix Q such that $v^TQv \geq 0$ whenever $v > 0$. Then

$$\rho(M^{-1}N) \leq \rho(M_Q^{-1}N_Q)$$

where $A = M_Q - N_Q$.

Proof: Let $\lambda_{\min}(M^{-1}A)$ and $\lambda_{\min}(M_Q^{-1}A)$ denote the smallest eigenvalues of $M^{-1}A$ and of $M_Q^{-1}A$, respectively. Since $M^{-1}N$ has nonnegative elements, the Perron-Frobenius theorem states that its spectral radius is equal to its (algebraically) largest eigenvalue. Hence we have

$$\rho(M^{-1}N) = \rho(I - M^{-1}A) = 1 - \lambda_{\min}(M^{-1}A).$$

If $\bar{v}$ is the eigenvector corresponding to the largest eigenvalue of $M^{-1}N$, then the Perron-Frobenius theorem also states that the elements of $\bar{v}$ are positive. Hence $\lambda_{\min}(M^{-1}A)$ also satisifes

$$\lambda_{\min}(M^{-1}A) = \frac{\bar{v}^TA\bar{v}}{\bar{v}^TM\bar{v}} \geq \frac{\bar{v}^TA\bar{v}}{\bar{v}^TM\bar{v} + \bar{v}^TQ\bar{v}} \geq \min_{v \neq 0} \frac{v^TAv}{v^TM_Qv} = \lambda_{\min}(M_Q^{-1}A),$$

and from this the desired result follows:

$$\rho(M_Q^{-1}N_Q) = \rho(I - M_Q^{-1}A) \geq 1 - \lambda_{\min}(M_Q^{-1}A) \geq 1 - \lambda_{\min}(M^{-1}A) = \rho(M^{-1}N). \qquad \square$$

Other work on iterative methods has focussed on preconditioners that are "optimal" in a different sense from that being considered here [e.g., 2]. Multigrid methods, for example, are "optimal" in the sense that the preconditioned matrix has condition number $O(1)$, independent of the size mesh from which the linear system was obtained (assuming that the linear system comes from a finite element approximation to an elliptic partial differential equation with sufficiently smooth solution). In this paper, to be considered "optimal", a preconditioner of the form of the multigrid preconditioner, must not only give a condition number $O(1)$, but the constant must be as small as possible, too.

3. Experimental Results for the 5-Point Laplacian.

In the following sections we report experimental results using the Overton optimization code [20] to find the matrix M of a given form which minimizes $\rho(I - L^{-1}ML^{-T})$, for a given matrix $A = LL^T$. According to theorems 1 and 2, an appropriate constant multiple of this matrix also minimizes $\rho(I - M^{-1}A)$ and $\kappa(M^{-1}A)$ over all matrices M of the given form. The matrix A was taken to be the 5-point Laplacian on a square with Dirichlet boundary

conditions:

$$A = \begin{bmatrix} D_1 & E_1 & & & \\ E_1^T & D_2 & & & \\ & & & E_{m-1} & \\ & & E_{m-1}^T & D_m & \end{bmatrix}, \tag{3.1}$$

$$D_i = \begin{pmatrix} 4 & -1 & & \\ -1 & & & \\ & & & -1 \\ & & -1 & 4 \end{pmatrix}, \quad E_i = \begin{pmatrix} -1 & & \\ & & \\ & & -1 \end{pmatrix} \quad ,i=1,\ldots,m.$$

In most cases, the matrix M was taken to have a fixed sparsity pattern; e.g., diagonal or tridiagonal. The matrix $I-L^{-1}ML^{-T}$ can then be expressed in the form

$$I - L^{-1}ML^{-T} = I - \sum_k \sum_q M_{kq} V^{(k,q)} ,$$

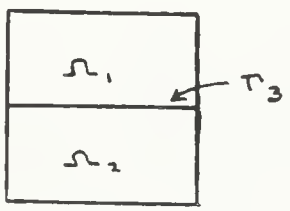
where the sum is over all indices (k,q) such that M_{kq} is allowed to be nonzero, and the matrices $V^{(k,q)}$ are given by

$$(V^{(k,q)})_{ij} = (L^{-1})_{ik} (L^{-1})_{jq} , \quad i,j=1,\ldots,n.$$

The matrices I and $-V^{(k,q)}$, $(k,q)\epsilon$ {indices of free elements of M} are the input to the optimization code.

In some cases, slightly different forms for M were considered. For example, in one experiment, we found the matrix M^{-1} having the same sparsity pattern as A, for which $\rho(I-L^TM^{-1}L)$ was minimal. This is equivalent to minimizing $\rho(I-M^{-1}A)$, and from Theorem 2, this also minimizes $\kappa(M^{-1}A)$ over all matrices M^{-1} with the given sparsity pattern. The elements of $L^TM^{-1}L$ are again linear functions of the free elements of M^{-1}, and so $\rho(I-L^TM^{-1}L)$ can be minimized with the same optimization code.

Another preconditioning problem considered was one involving not A but the Schur complement C in A of a block corresponding to a dividing line in the center of the square domain:



If nodes in Ω_1 are numbered first, then nodes in Ω_2, and finally nodes on the boundary Γ_3, then the matrix A takes the form

$$A = \begin{bmatrix} K_{11} & 0 & K_{13} \\ 0 & K_{22} & K_{23} \\ K_{13}^T & K_{23}^T & K_{33} \end{bmatrix} . \tag{3.2}$$

The Schur complement of K_{33} in A is defined as

$$C = K_{33} - K_{13}^T K_{11}^{-1} K_{13} - K_{23}^T K_{22}^{-1} K_{23} , \tag{3.3}$$

and the problem of solving a linear system with coefficient matrix A can be reduced to one of solving a smaller linear system with coefficient matrix C. This is the basis of many domain decomposition methods [3]. One still needs a good preconditioner for the matrix C, and here we considered preconditioners with a given sparsity pattern; e.g., tridiagonal, as well as certain other forms; e.g., Toeplitz. Again, the matrix C was factored in the form LL^T, and the function $\rho(I - L^{-1}ML^{-T})$ was minimized. This is a convex function of the nonzero elements of M (when M is restricted to have a certain sparsity pattern) or of the values on each diagonal of M (when M is restricted to be Toeplitz).

Many of the most efficient preconditioners M are easy to solve not because they themselves have a special sparsity pattern, but because they are of the form KK^T, where K is a lower triangular matrix with a simple sparsity pattern; e.g., that of the lower triangle of A. The problem of finding the matrix K of a given form which minimizes $\rho(I - L^{-1}KK^TL^{-T})$ is more difficult than the previously described problems, however, because the matrix

$$I - L^{-1}KK^TL^{-T} \tag{3.4}$$

is not an affine function of the elements of K. Moreover, its spectral radius is not a convex function of K, and the function may have local minima.

The optimization code is easily modified to handle the case of matrix elements which are nonlinear functions of the unknowns, but there is no guarantee that the solution it finds will be the *global* minimum. Still, it will be shown in the numerical examples that the optimization code is able to find preconditioners of the form (3.4), where K has a fixed sparsity pattern, that are significantly better than many currently used preconditioners. We considered matrices K having the same sparsity as the lower triangle of A, and compared the preconditioner KK^T returned by the optimization code with the incomplete Cholesky decomposition [19], the modified incomplete Cholesky decomposition [15], and the SSOR preconditioner [25]. We also considered matrices K having the same sparsity pattern as the lower triangular factor of the hierarchical basis function preconditioner [26] and compared the condition number for this algorithm with the locally minimized condition number returned by the optimization code.

Experiments that have been carried out so far are for very small problems. It is planned to continue this work on larger problems when the optimization code has been ported to larger and faster machines. It should be stressed that this is not meant as a practical means for finding a good preconditioner for a given problem. It is much easier to solve the linear system than it is to find the optimal preconditioner of a given class. Rather, the optimization code is meant to provide insight into the properties of preconditioners. If the optimal preconditioner of a certain form for a given problem does not give rise to a preconditioned matrix with small condition number, then it is not worthwhile considering preconditioners of that form (unless such preconditioners can exhibit other desirable properties, such as tight clustering of most of the eigenvalues). On the other hand, if the code shows that there is a good preconditioner of the given form, then it still may or may not be possible to compute such a preconditioner in a reasonable amount of time.

3.1. Diagonal Preconditioning for A.

As a check on the code, we first tried computing the optimal *diagonal* preconditioner for the matrix A of (3.1). According to the Forsythe and Strauss theorem of Section 2, this is simply $D = diag(A)$. Indeed, even from very far off initial guesses, the code always converged to $M = diag(A)$ and indicated that it had successfully found the minimum. This gave us confidence to try problems for which the answers were not known.

3.2. Tridiagonal Preconditioning for A.

By numbering the odd block-rows and block-columns first and the even ones last, the matrix A of (3.1) can be permuted into block 2-cyclic form, without changing the diagonal blocks. It follows from the Eisenstat, Lewis, Schultz theorem of Section 2, that the optimal block diagonal preconditioner for A is $D = block\ diag(A)$. This matrix is also tridiagonal. The optimization code was used to compute the best *tridiagonal* preconditioner for A, and it was found to be slightly better than the block diagonal matrix D. Fig. 1 shows condition numbers for A, $D^{-1}A$, and $M^{-1}A$, where M is the optimal tridiagonal preconditioner, plotted against h^{-2}. From the figure, it appears that all of these matrices have condition number $O(h^{-2})$, and based on these results we make the following conjecture:

Conjecture 1. Let A_h be the 5-point Laplace matrix of (3.1), for grid size h, and let M_h be any symmetric positive definite tridiagonal preconditioner for A_h. Then the condition number $\kappa(M_h^{-1}A_h)$ of the preconditioned matrix satisfies

$$\kappa(M_h^{-1}A_h) \geq O(h^{-2}) .$$

Based on these results we conclude that tridiagonal preconditioners for A cannot be very effective, in terms of giving a small condition number, for small values of h. Additional experiments were performed to find optimal preconditioners of slightly larger bandwidths, but again they appeared to give condition numbers for the preconditioned matrix that were $O(h^{-2})$.

The elements of the optimal tridiagonal preconditioner for different values of h are listed in Table 1. The tridiagonal matrix given in the table has been multiplied by an appropriate constant so that it minimizes $\rho(I-M^{-1}A)$ (although the code actually computes the tridiagonal matrix which minimizes $\rho(I-A^{-1}M)$). Note that the preconditioners are persymmetric (symmetric about their northeast to southwest diagonal), as would be expected due to the symmetry of the problem. This restriction was not enforced on the class of matrices from which the optimal one was to be found, and so it gives further evidence that the optimization code is functioning correctly. Also given in Table 1 are some of the eigenvalues of the preconditioned iteration matrices $I-M^{-1}A$. Note that for all problem sizes, the largest and smallest eigenvalues of the preconditioned iteration matrix each have multiplicity 2. Based on this evidence we make the following conjecture:

Conjecture 2. Let A_h be the 5-point Laplace matrix of (3.1) for grid size h, and let M_h be the optimal tridiagonal preconditioner for A_h. If $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{n-1} \leq \lambda_n$ are the eigenvalues of $M_h^{-1}A_h$, then the largest and smallest eigenvalues satisfy

$$\lambda_1 = \lambda_2 , \quad \lambda_{n-1} = \lambda_n .$$

h = 1/5		h = 1/6		h = 1/7	
eigenvalues -0.634, -0.634, -0.255, -0.230, 0.020, 0.070,... 0.314, 0.375, 0.437, 0.487, 0.634, 0.634		eigenvalues -0.732, -0.732, -0.406, -0.400, -0.034, -0.031,... 0.507, 0.575, 0.593, 0.610, 0.732, 0.732		eigenvalues -0.798, -0.798, -0.539, -0.508, -0.188, -0.182,... 0.671, 0.679, 0.692, 0.699, 0.798, 0.798	
diagonal	subdiagonal	diagonal	subdiagonal	diagonal	subdiagonal
4.4111	-1.2754	4.5383	-1.3198	4.6009	-1.3366
4.7698	-1.6842	5.1029	-1.8296	5.2721	-1.8816
5.0153	-1.8785	5.7456	-2.3460	6.2098	-2.5799
5.2966	-1.1143	6.1396	-2.4573	7.0826	-3.0903
4.9314	-1.5875	6.3366	-1.4563	7.5027	-3.1271
4.9285	-1.6675	5.6482	-1.9397	7.5623	-1.8487
4.9806	-1.7323	5.6718	-2.0424	6.4873	-2.3507
5.1583	-1.2442	5.8645	-2.2516	6.4556	-2.4075
5.1583	-1.7323	6.0232	-2.2930	6.7498	-2.7103
.	.	6.2313	-1.7199	7.1391	-2.9591
.	.	6.0631	-2.1707	7.3354	-2.9480
.	.	5.9260	-2.1821	7.4991	-2.2456
		5.9285	-2.1821	7.1410	-2.6953
		5.9260	.	7.0021	-2.7208
		.	.	7.0845	-2.8095
		.	.	7.1929	-2.8771
		.		7.2359	-2.8434
				7.3990	-2.3548
				7.3990	-2.8434
				.	.
				.	.
				.	.

Table 1. Optimal Tridiagonal Preconditioners for A.

3.3. Approximation to A^{-1} with Same Sparsity as A.

Upon computing the inverse of the matrix A of (3.1), one finds that the larger elements of the inverse are on or near diagonals in which A has nonzeros. It therefore seems reasonable to consider an approximation M^{-1} to A^{-1}, which has nonzero entries only in the diagonals where A has nonzeros. If an effective preconditioner of this form could be found, it would result in a highly vectorizable or parallelizable algorithm, since "solving" the preconditioner would now just mean multiplying by a sparse matrix. The optimization code was used to find the matrix M^{-1} having nonzeros only in the five diagonals where A has nonzeros, for which $\kappa(M^{-1}A)$ was as small as possible.

This turned out to be a more difficult problem for the optimization code, than either of the previous two cases. Using a very coarse mesh, $h = 1/3$, the optimization code returned a solution with the following caveat

Apparently optimal (or very nearly optimal) with non-unique solution
(since a Lagrange matrix nearly singular)

The eigenvalues of the matrix $I-M^{-1}A$, for the computed matrix M^{-1}, were all equal in magnitude, with three being positive and one negative. Upon restarting the code from a different initial guess, it returned with the same warning message but a different optimal M^{-1}. For this newly computed matrix M^{-1}, the eigenvalues of $I-M^{-1}A$, were again all equal in magnitude, and of the same magnitude as those previously computed, but this time three were negative and one positive. The elements of the two different optimal matrices M^{-1} are given in Table 2.

For smaller values of h, the problem appears to have a unique solution. The code was able to find and identify as such the optimal M^{-1}, for $h=1/5$. For $h=1/4$, 1/6, and 1/7, attempts at finding the minimum resulted in the code halting with the message

radius too small

indicating that its trust region radius had been reduced below the machine precision and it had been unable to find a descent direction. Restarting, in the case $h=1/4$, resulted in the code finding approximately the same "solution", but this time giving the message "Apparently optimal (or very nearly optimal) with non-unique solution." In the case $h=1/6$, several restarts resulted in the code finding approximately the same "solution", but still halting because the trust region radius was too small. A restart in the case $h=1/7$, however, resulted in the code finding a significantly different approximation and identifying it as optimal. The spectral radius of $I-M^{-1}A$ for this "truly" optimal M^{-1} was .680, compared to .685 for the M^{-1} at which it stopped the first time because of a too small radius. It is believed that the spectral radii returned by the code, for the different sizes of h, are all near optimal, though the actual matrices M^{-1} may be significantly further from the optimal ones.

An approximation to A^{-1} that has been suggested in the literature [8,16] is the following. If we write A in the form

$$A = D^{1/2}(I-G)D^{1/2}$$

where D is a diagonal matrix, and if the spectral radius of G is less than 1 (which it is for this problem), then A^{-1} is given by

$$A^{-1} = D^{-1/2}(I+G+G^2+\ldots)D^{-1/2}.$$

An approximation to A^{-1} is obtained by retaining just a finite number of terms in the infinite Neumann series above. The approximation can be improved by multiplying each term retained by an appropriate constant [16]. If only one term is retained, then this approximation has the same sparsity pattern as A. Thus, A^{-1} can be approximated by

$$M_1^{-1} \equiv D^{-1/2}(c_0 I + c_1 G)D^{-1/2}. \tag{3.3.1}$$

When A is the 5-point Laplacian, the optimal constants are just $c_0=c_1=1$.

The condition number of A, that of $M_1^{-1}A$, where M_1^{-1} is defined by (3.3.1), and that of $M^{-1}A$, where M^{-1} is the optimal preconditioner having nonzeros only in diagonals where A has nonzeros (or, at least, the preconditioner returned by the optimization code) are plotted in Fig. 2. Note that the condition number of $M_1^{-1}A$ is almost the same as that of $M^{-1}A$ for

all values of h! It should be noted, however, that in cases where the code was able to find a matrix which it identified as optimal, it was significantly different from the matrix in (3.3.1). The condition number of the preconditioned matrix was very nearly the same, but the eigenvalues of the optimally preconditioned matrix tended to cluster somewhat more at the ends and less in the middle. For the $h=1/5$ case, for example, for the optimal M^{-1}, the matrix $I-M^{-1}A$ had 5 eigenvalues equal to -.474, 3 eigenvalues equal to +.474, and no repetitions among the remaining interior eigenvalues. Using the matrix of (3.3.1), $I-M_1^{-1}A$ had 4 eigenvalues equal to -.486, 4 equal to -.393, 2 equal to -.344, 4 equal to -.022, and 2 equal to +.486.

It is known that a polynomial preconditioner of the form (3.3.1) can reduce the number of conjugate gradient steps (over the number required with no preconditioner) by no more than a factor of 2 (one plus the degree of the polynomial in G) [8]. It is also known that when A is the 5-point Laplacian, for small values of h, the condition number of the matrix $M_1^{-1}A$ is approximately 1/4 that of A. This is not apparent from Fig. 2, however, because only very coarse grid sizes are shown. Because the asymptotic behavior of preconditioner (3.3.1) cannot be predicted from the figure, we are wary of predicting the behavior of the optimally preconditioned matrix, based on these results. From the figure, it would appear that the condition number of the optimally preconditioned matrix is still $O(h^{-2})$, but testing on larger problems is needed to see if this trend continues.

eigenvalues -1.429, 1.429, 1.429, 1.429			eigenvalues -1.429, -1.429, -1.429, 1.429		
diag	1st subdiag	2nd subdiag	diag	1st subdiag	2nd subdiag
2.857e-01	7.143e-02	7.143e-02	2.857e-01	7.143e-02	7.143e-02
2.500e-01	3.571e-02	7.143e-02	3.214e-01	3.571e-02	7.143e-02
2.500e-01	7.143e-02		3.214e-01	7.143e-02	
2.857e-01			2.857e-01		

Table 2. Two "Optimal" Matrices M^{-1} with Same Sparsity Pattern as A ($h=1/3$).

One experiment was performed in which the approximation to A^{-1} was allowed to have nonzeros in extra diagonals -- the same diagonals in which the second degree polynomial preconditioner

$$M_2^{-1} = c_0 I + c_1 G + c_2 G^2$$

has nonzeros. In this case, the code again stopped with the message "radius too small," so it is not known how close it came to finding the optimal preconditioner of the given sparsity pattern. Still, the matrix M^{-1} returned by the optimization code was considerably better than the polynomial preconditioner M_2^{-1} with the optimal coefficients, c_0, c_1, c_2. For $h=1/5$, the spectral radius of $I-M^{-1}A$ was .208, compared to .260 for $I-M_2^{-1}A$.

3.4. Preconditioners for the Schur Complement C.

In this set of experiments we found optimal preconditioners of various forms for the Schur complement matrix C defined in (3.3). The condition number of the matrix C is $O(h^{-1})$ and, in fact, its entire eigendecomposition can be derived using Fourier analysis [3]. Based on this, one can determine analytically the optimal *Toeplitz tridiagonal* preconditioner [24,4] (since any Toeplitz tridiagonal matrix has the same eigenvectors as C), and it can be shown that this preconditioned matrix has condition number $O(h^{-1/2})$. With the optimization code, we were able to determine the optimal tridiagonal preconditioner, without requiring that it be Toeplitz. This turned out to be a relatively easy problem for the optimization code, with the code returning a solution which it identified as optimal, for all values of h tested. It was observed, however, that especially for the smaller values of h, the optimal tridiagonal preconditioner for C was very nearly Toeplitz. In Fig. 3 are plotted the condition numbers of C, of $K_{33}^{-1}C$, where K_{33} is as defined in (3.2), of $T^{-1}C$, where T is the optimal Toeplitz tridiagonal preconditioner for C, and of $M^{-1}C$, where M is the optimal tridiagonal preconditioner for C, returned by the optimization code. Note that the curves for $\kappa(T^{-1}C)$ and for $\kappa(M^{-1}C)$ are almost identical. In Fig. 3b, these two curves only are plotted versus $h^{-1/2}$. Based on this evidence, we make the following conjecture:

Conjecture 3. Let C_h be the Schur complement matrix corresponding to a center dividing line for the 5-point Laplace matrix on a grid of size h, as defined in (3.3). Let M_h be any symmetric positive definite tridiagonal preconditioner for C_h. Then the condition number $\kappa(M_h^{-1}C_h)$ of the preconditioned matrix satisfies

$$\kappa(M_h^{-1}C_h) \geq O(h^{-1/2}) .$$

The next experiment was to find the optimal (dense) Toeplitz preconditioner for C. This turned out to be a very difficult problem for the optimization code because of very tight clustering of many eigenvalues at the extremes. The first attempt at solving this problem for most values of h resulted in the code halting with the message "radius too small," as described previously. Restarting the code with slightly different options, however, resulted in its identifying essentially the same "solution" as optimal.

The condition number of the optimally preconditioned matrix is plotted in Fig. 4, along with that of the matrix preconditioned by a Toeplitz preconditioner proposed by Golub and Mayers [12]. Although there is a large relative difference in these condition numbers, both are so small that this difference is not really significant. The eigenvalues of both preconditioned matrices are plotted in Figs. 4b-c, for the case $h = 1/16$. For both preconditioners there is considerable clustering of the eigenvalues, but it is especially pronounced for the Toeplitz preconditioner returned by the optimization code.

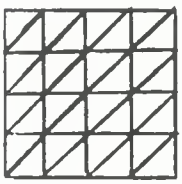
3.5. Preconditioners of the Form KK^T, Where K Has the Same Sparsity as the Lower Triangle of A.

Several well-known preconditioners -- e.g., the incomplete Cholesky (IC) and modified incomplete Cholesky (MIC) decompositions [19,15], and the symmetric successive overrelaxation (SSOR) preconditioner [25] -- are of the form KK^T, where K is a lower triangular matrix with the same sparsity pattern as the lower triangle of A. As mentioned earlier, the problem of finding the optimal preconditioner of this form is not one of minimizing a convex function, and so the "solution" returned by the optimization code may not be the global minimum. The code can, however, be used to find a local minimum for the spectral radius of (3.4), when K is restricted to have the same sparsity pattern as the lower triangle of A. This also turned out to be a difficult problem for the optimization code, and in all cases, it reached a

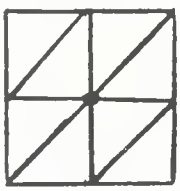
point at which it could go no further because its trust region radius had been reduced below the precision of the machine. Still, the preconditioner KK^T returned by the optimization code was significantly better than those proposed in the literature. Condition numbers for the matrix preconditioned by the IC decomposition, the MIC decomposition, and the SSOR preconditioner are plotted in Fig. 5, along with that of the matrix preconditioned by KK^T, where K is the matrix returned by the optimization code. It remains to be seen how much further this condition number can be reduced.

3.6. Preconditioners of the Form KK^T, Where K has the Same Sparsity as the Lower Triangular Factor of the Hierarchical Basis Function Preconditioner.

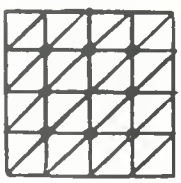
A recently proposed idea for reducing the condition number of the matrix A of a finite element approximation is to use different basis functions to represent the finite element space. One such idea -- the hierarchical basis function representation [26] -- yields a matrix $\hat{A}$ whose condition number is $O(\log h^{-1})^2$ rather than $O(h^{-2})$ for the standard nodal basis representation. This idea can also be thought of as solving the original nodal basis matrix A using a special preconditioner. The preconditioner is of the form KK^T, where, if nodes are numbered appropriately, K is a lower triangular matrix with a certain sparsity pattern based on the hierarchy of grids and the type of finite element. The 5-point operator A that we are considering corresponds to a linear finite element approximation on a regular triangular grid.



A hierarchy of grids and basis functions for the space of continuous piecewise linear functions on the given grid can be defined as follows. Start with a coarsest grid which has only one interior node and take the first basis function to be linear in each zone of this coarsest grid and to have value one at the interior node and zero at all the boundary nodes:



Divide each triangle of this grid into four congruent triangles by connecting the midpoints of each side. For each new interior node, take a basis function that is linear in each zone of this finer grid and has value one at that node and zero at all other nodes of this finer grid.



Create an even finer grid by connecting the midpoints of the sides of each of these triangles and adding basis functions similarly for each new interior node. Continue, until the finest level grid is reached. Any continuous piecewise linear function on this finest grid can be represented as a linear combination of these hierarchical basis functions,

$$v(x,y) = \sum_{j=1}^{n} \beta_j \psi_j(x,y) , \qquad \psi_j(x,y) = \textit{hierarchical basis function which is 1 at node } j,$$

just as it can be represented as a (different) linear combination of the usual nodal basis functions (piecewise linear functions which are one at one node and zero at all other nodes of the *finest* grid),

$$v(x,y) = \sum_{j=1}^{n} \alpha_j \phi_j(x,y) , \qquad \phi_j(x,y) = \textit{nodal basis function which is 1 at node } j.$$

Solving the matrix K^T means translating from hierarchical to nodal basis coefficients. That is, if a vector β contains the coefficients of a piecewise linear function with respect to the hierarchical basis functions then $\alpha = K^{-T}\beta$ is the vector of coefficients of the same piecewise linear function with respect to the usual nodal basis functions. To solve the linear system $K^T\alpha = \beta$, one starts with the coarsest grid and takes the component of α at that point to be the same as the corresponding component of β. One then moves to the next finer grid and takes the component of α at each new node to be the component of β at that node plus the average of the components of α at each of the surrounding coarser grid nodes. Continuing in this way to the finest grid, one obtains the vector of coefficients of the nodal basis functions. It follows that if the coarsest grid point is numbered last, preceded by those on the next finer grid, etc., with the points that lie only on the finest grid numbered first, then K^T is an upper triangular matrix with nonzero elements on its main diagonal and in positions corresponding to the couplings between nodes on one grid level and their nearest neighbors on the next *coarser* grid level. Note that the couplings are *not* between nearest neighbors on the finest grid, as they are in the nodal basis matrix A.

One might ask how small a condition number can be obtained with a preconditioner of this form. Again, since it is the sparsity pattern of the triangular factors that is fixed, the optimization code cannot necessarily find the optimal preconditioner of this form. It can, however, give an *upper* bound on the minimal condition number.

When the triangular factors K and K^T are multiplied together, they produce a preconditioner M which can have nonzeros only in certain positions. (M_{ij} can be nonzero only if, for some k, K_{ik} and K_{jk} are both nonzero.) One might also ask how small a condition number can be obtained with a preconditioner M having this sparsity pattern. Clearly, this gives a *lower* bound for the minimal condition number obtainable from triangular factors having the sparsity of K and K^T. The optimization code can be used to solve this problem, and since the function to be minimized is convex, if the code finds a minimum, then it is the global minimum.

In Table 3, the condition number of the matrix preconditioned by the hierarchical basis preconditioner is listed, along with upper and lower bounds on the minimal condition number obtainable from a preconditioner of the form KK^T, where K has the same sparsity pattern as the lower triangular factor of the hierarchical basis preconditioner. To avoid ambiguity in defining the hierarchy of grids, we considered only grid sizes which are powers of 2; i.e., $h = 1/4$ and $h = 1/8$. For the $h = 1/8$ problem, when the sparsity of M was specified (to obtain a lower bound on the minimal condition number when the sparsity of K is specified), the optimization code did not find a point which it identified as optimal. It stopped because of a too small trust region radius, and attempts at restarting resulted in its finding approximately the same "solution", but again failing to identify it as optimal. Hence, this entry in the table is considered somewhat uncertain. Although the optimization code shows that the hierarchical basis function preconditioner can clearly be improved upon for the grid sizes shown, this is of questionable importance. It is only for finer grids that the hierarchical basis function preconditioner is really expected to be efficient. At these coarse grid sizes, the condition

number of the preconditioned matrix is growing much faster than $O(\log h^{-1})^2$, so it is not clear that improvements seen at these coarser grid levels will also be possible for finer grids. Unfortunately, it is currently impractical to use the optimization code for finer grid sizes.

		$\kappa(L^{-1}KK^TL^{-T})$, K from optimization code	$\kappa(L^{-1}ML^{-T})$, M from optimization code
h^{-1}	Hierarchical Basis Cond. No.	Upper Bound on Minimal Cond. No.	Lower bound on Minimal Cond. No.
4	4.56	3.00	3.00
8	10.59	5.27	4.86 ?

Table 3. Condition Number for Hierarchical Basis Preconditioner and Upper and Lower Bounds on Condition Number for the Optimal Preconditioner of the Same Form.

4. Experimental Results for Other Diffusion-Type Operators.

To determine if other diffusion-type elliptic operators could be preconditioned more or less effectively than the Laplacian, we attempted to compute optimal preconditioners of some of these same forms for several other operators. The problems were all of the form

$$\nabla\cdot\rho\nabla u = f, \quad \textit{on } (0,1)\ x\ (0,1),$$

$$u(x,0) = u(x,1) = u(0,y) = u(1,y) = 0,$$

where the diffusion coefficient ρ was varied. The values of ρ considered were:

$$\rho(x,y) = 1. \quad (\textit{as in the previous section}) \tag{a.}$$

$$\rho(x,y) = .01 + x^2 + y^2. \tag{b.}$$

$$\rho(x,y) = \begin{cases} 1 & \text{if } x < .5 \\ 100 & \text{if } x \geq .5 \end{cases} \tag{c.}$$

$$\rho(x,y) = \begin{cases} 1 & \text{if } x < .5 \text{ and } y < .5 \\ 10^4 & \text{if } x \geq .5 \text{ and } y < .5 \\ 10^{-4} & \text{if } x < .5 \text{ and } y \geq .5 \\ 5 & \text{if } x \geq .5 \text{ and } y \geq .5 \end{cases} \tag{d.}$$

To avoid ambiguity in differencing around the discontinuities in ρ, a continuous piecewise

linear finite element approximation was used to generate the matrix A corresponding to each of these operators.

4.1. Diagonal Preconditioners for A.

Although a highly-varying diffusion coefficient, such as that in (d), results in a very badly conditioned finite difference or finite element matrix, it has long been known that a simple diagonal scaling could greatly reduce this condition number. In Fig. 6 are plotted the condition numbers of the matrices for these four problems, after they have been scaled by their diagonals. According to the Forsythe&Strauss theorem, this is the optimal diagonal preconditioner for these matrices. Note from the figure that after diagonal scaling, the condition numbers for all of these matrices are nearly the same, with case (b) actually being somewhat better conditioned than the Laplacian (a). In cases (c) and (d), this condition number jumps slightly when the mesh size $1/h$ is not even. In this case, the finite element equations are a poor approximation to the differential equation, anyway, because the discontinuity of ρ, and hence ∇u, occurs within a mesh cell.

It is also interesting to note that, although the optimal diagonal preconditioners for these problems are known, the optimization code had a great deal of difficulty in finding the solutions for cases (c) and (d). For the $h = 1/4$ grid, starting from an initial guess that was equal to twice the true solution, the code required 272 iterations to find the optimal diagonal preconditioner for case (c), compared to just 12 for case (b). In case (d), the code was stopped after 600 iterations when only negligible improvement had been made over the initial guess. The reasons for this difficulty are currently being investigated to determine if some rescaling or other modification of the problem can make the optimization job easier.

4.2. Tridiagonal Preconditioners for A.

Because a simple diagonal scaling results in matrices that are about equally well-conditioned, one might expect that any class of preconditioners which is broad enough to contain all diagonal matrices, e.g., tridiagonal preconditioners, would be about as effective for one of these operators as for another. This did not turn out to be the case, however. The optimal tridiagonal preconditioner in cases (a) and (b) gave a significantly smaller condition number than in cases (c) and (d). The optimization code again had difficulty in finding the optimal preconditioners for cases (c) and (d), but after many iterations it finally did find solutions which it identified as optimal. These condition numbers are plotted in Fig. 7. The point $h = 1/5$ was not included in cases (c) and (d), since the difference equation is not a good approximation to the differential equation in this case (since the discontinuity of ρ occurs within a zone). For grids containing mesh lines along the discontinuities of ρ, we expect that the condition number for all of these optimally preconditioned operators will grow as $O(h^{-2})$.

4.3. Preconditioners of the Form $D\Delta D$, Where Δ is the Laplacian and D is Diagonal.

It was recently proved that the matrix A from an arbitrary second-order self-adjoint elliptic partial differential equation can be preconditioned by the matrix Δ corresponding to the Laplacian on the same grid and with the same boundary conditions so that the resulting preconditioned matrix has condition number $O(1)$, independent of the mesh size [17]. Since the Laplacian is relatively easy to solve on a rectangular grid (and with an integral equation formulation, perhaps also on an irregular grid [18]), this might make an effective preconditioner. Unfortunately, however, the constant in the condition number bound can be quite large, and it is large for problems (c) and (d). One might ask if there is a simple modification of the Laplacian that would still be easy to solve and would give a condition number that is not only $O(1)$ but has a small constant as well. The simplest idea is to scale the Laplacian

by some diagonal matrix D and use the symmetric matrix $D\Delta D$ as the preconditioner. This is the same form of preconditioner that was studied by Concus and Golub [5], who determined the optimal such preconditioner for 1-D problems.

We used the optimization code to find the diagonal matrix D for which the spectral radius of $I - L^{-1}D\Delta DL^{-T}$ was as small as possible, where $A = LL^T$, was the matrix arising from problems (b-d). Again, the optimization code had difficulty with problem (d), so we omitted this from our results. In Fig. 8 are plotted the condition numbers for problems (b) and (c), preconditioned by the optimally scaled Laplacian-type preconditioner, $D\Delta D$. Computations for case (c) were performed on somewhat finer grids than for case (b), as will be explained below. Although the condition number for each of these preconditioned matrices is bounded by a constant independent of h, we have not reached a small enough value of h to determine this constant. Clearly, problem (b) is very well-conditioned by the appropriate diagonal scaling of the Laplacian, and problem (c) is also reasonably well approximated by such a preconditioner, for the grid sizes shown.

The optimal diagonal matrix determined in problem (c) was especially interesting. If nodes to the left of the discontinuity in ρ are numbered first, then nodes to the right of the discontinuity, and finally nodes on the discontinuity, then the optimal diagonal scaling for the Laplacian-type preconditioner, as determined by the optimization code, has the following form:

$$D = \begin{pmatrix} d_1 I & & \\ & d_2 I & \\ & & d_3 I \end{pmatrix},$$

where d_i, $i = 1,2,3$ are constants and each block corresponds to one of the above mentioned subregions. Using this ordering of nodes, the matrix A arising from problem (c) has the form

$$A = \begin{bmatrix} c_1 K_{11} & 0 & c_1 K_{13} \\ 0 & c_2 K_{22} & c_2 K_{23} \\ c_1 K_{13}^T & c_2 K_{23}^T & c_3 K_{33} \end{bmatrix},$$

where the blocks $K_{i,j}$, $i,j = 1,2,3$ are the blocks of the Laplacian, and the constants c_i, $i = 1,2,3$ are

$$c_1 = 1, \quad c_2 = 100, \quad c_3 = 50.5 .$$

The preconditioner $D\Delta D$ returned by the optimization code then has the form

$$M = \begin{bmatrix} d_1^2 K_{11} & 0 & d_1 d_3 K_{13} \\ 0 & d_2^2 K_{22} & d_2 d_3 K_{23} \\ d_1 d_3 K_{13}^T & d_2 d_3 K_{23}^T & d_3^2 K_{33} \end{bmatrix}. \tag{4.3.1}$$

The constants d_i, $i = 1,2,3$, are listed in Table 4, for grid sizes $1/h = 4,6,8$. Seeing this pattern, we were able to go to finer grid sizes by restricting the preconditioner to be of the form (4.3.1), and having the optimization code determine only the best constants d_i, $i = 1,2,3$. These are also listed in Table 4, for $1/h = 10,16$, and the corresponding condition numbers are plotted in Fig. 8. The condition number of the preconditioned matrix still has not reached its

asymptotic limit, however, and we do not know what this limit is. It is interesting to note that at grid size $1/h=16$, d_2 is equal to d_3. The difference between d_1 and d_2 or d_3 is also becoming smaller and probably approaches some asymptotic limit.

$1/h$	d_1	d_2	d_3
4	1.97	16.92	13.52
6	1.94	14.60	12.77
8	1.91	13.05	12.08
10	1.89	11.89	11.49
16	1.88	10.05	10.05

Table 4. Constants Defining the Optimal Preconditioner of the Form $D\Delta D$.

5. Conclusions.

We have demonstrated a very useful tool in the study of preconditioners. Again, an optimization code is not usually a practical method for finding a good preconditioner for a given problem, but rather it is intended to give insight into the properties of preconditioners and the forms of matrices that can or cannot be potentially effective preconditioners. Results from the code have led to several conjectures and may also contribute to the ultimate goal of finding an easily solved preconditioner that gives a condition number that is $O(1)$ and has a small constant factor, for a large class of matrices arising from elliptic partial differential equations.

Acknowledgments.

The authors would like to thank Michael Overton for his work in developing his optimization code, advising us on its use, and refining its capabilities in response to users' needs. Olof Widlund, Tony Chan, and Gene Golub also contributed significantly through helpful discussions.

References:

[1] P. E. Bjorstad and O. B. Widlund, "Iterative Methods for the Solution of Elliptic Problems on Regions Partitioned into Substructures," *SIAM Jour. Num. An. 23, #6,* 1097-1120, Dec., 1986.

[2] A. Brandt, "Multilevel Adaptive Solutions to Boundary Value Problems," *Math. of Comp. 31* 333-390, 1977.

[3] T. F. Chan, "Analysis of Preconditioners for Domain Decomposition," *SIAM Jour. Num. An. 24, #2,* 382-390, Apr., 1987.

[4] T. Chan, private communication.

[5] P. Concus and G. Golub, "Use of Fast Direct Methods for the Efficient Numerical Solution of Nonseparable Elliptic Equations," *SIAM J. Numer. Anal. 10, #6,* 1103-1120, December, 1973.

[6] J. Cosgrove, J. Diaz, and A. Griewank, "Approximate Inverse Preconditionings for Sparse Linear Systems," to appear.

[7] J. Demmel, "The Condition Number of Equivalence Transformations that Block Diagonalize Matrix Pencils," *SIAM J. Numer. Anal. 20, #3,* 599-610, June, 1983.

[8] P. F. Dubois, A. Greenbaum and G. H. Rodrigue, "Approximating the Inverse of a Matrix for Use in Iterative Algorithms on Vector Processors," *Computing 22,* 257-268, 1979.

[9] S. Eisenstat, J. Lewis, and M. Schultz, "Optimal Block Diagonal Scaling of Block 2-Cyclic Matrices," *Lin. Alg. Appl., 44,* 181-186, 1982.

[10] R. Fletcher, "A Survey of Algorithms for Unconstrained Optimization," Rep. TP456, Theoretical Physics Division, A.E.R.E. Harwell, Didcot, Berks, England, 1971.

[11] G. E. Forsythe and E. G. Strauss, "On Best Conditioned Matrices," *Proc. Amer. Math. Soc. 6,* 340-345, 1955.

[12] G. Golub and D. Mayers, "The Use of Pre-Conditioning over Irregular Regions," in *Computing Methods in Applied Sciences and Engineering, VI,* R. Glowinski and J. L. Lions, eds. Elsevier, North-Holland, 1984.

[13] A. Greenbaum, "Comparison of Splittings Used with the Conjugate Gradient Algorithm," *Num. Math. 33,* 181-194, 1979.

[14] A. Greenbaum, "Behavior of Slightly Perturbed Lanczos and Conjugate Gradient Recurrences," *Lin. Alg. Appl. 113,* 7-63, 1989.

[15] I. Gustafsson, "A Class of 1st Order Factorization Methods," *BIT 18,* 142-156, 1978.

[16] O. G. Johnson, C. A. Micchelli, and G. Paul, "Polynomial Preconditioning for Conjugate Gradient Calculations," *SIAM Jour. Num. An. 20, #2,* 362-376, 1983.

[17] T. A. Manteuffel and S. V. Parter, "Preconditioning and Boundary Conditions," LA-UR-88-2626, Los Alamos Technical Report, July, 1988.

[18] A. Mayo, "The Fast Solution of Poisson's and the Biharmonic Equations on Irregular Regions," *SIAM Jour. Num. An. 21, #2,* 285-299, Apr., 1984.

[19] J. A. Meijerink and H. A. Van der Vorst, "An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix is a Symmetric M-Matrix," *Math. Comp. 31,* 148-162, 1977.

[20] M. L. Overton, "On Minimizing the Maximum Eigenvalue of a Symmetric Matrix," *SIAM J. Mat. An. Appl., Vol. 9, #2,* 256-268, Apr., 1988

[21] C. Romine, private communication.

[22] A. van der Sluis, "Condition Numbers and Equilibration Matrices," *Num. Math. 14,* 14-23, 1969.

[23] R. S. Varga, *Matrix Iterative Analysis,* Prentice-Hall, Englewood Cliffs, N.J., 1962.

[24] O. Widlund, private communication.

[25] D. M. Young, *Iterative Solution of Large Linear Systems,* Academic Press, New York, 1971.

[26] H. Yserentant, "On the Multi-level Splitting of Finite Element Spaces," *Num. Math. 49,* 379-412, 1986.

Fig. 1. Tridiagonal Preconditioners for A

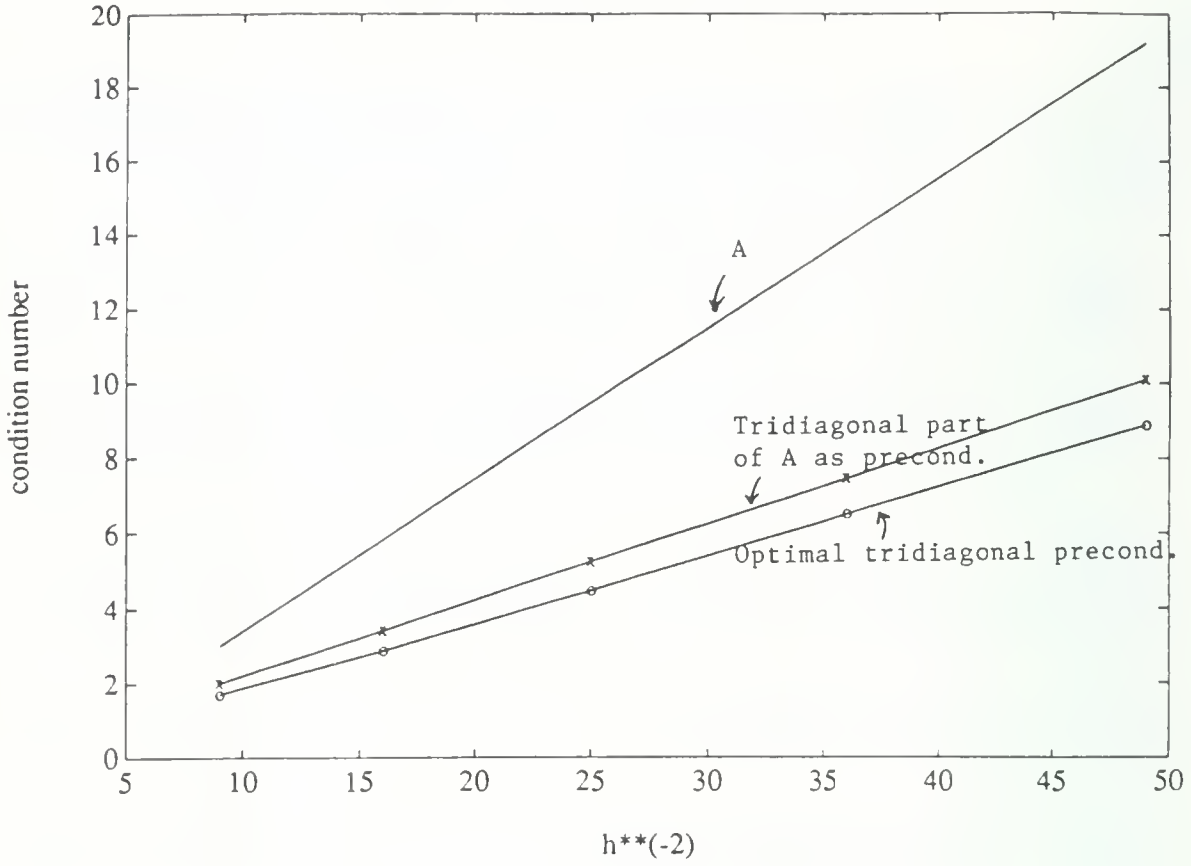


Fig. 2. Approximations to A-inv with Same Sparsity as A

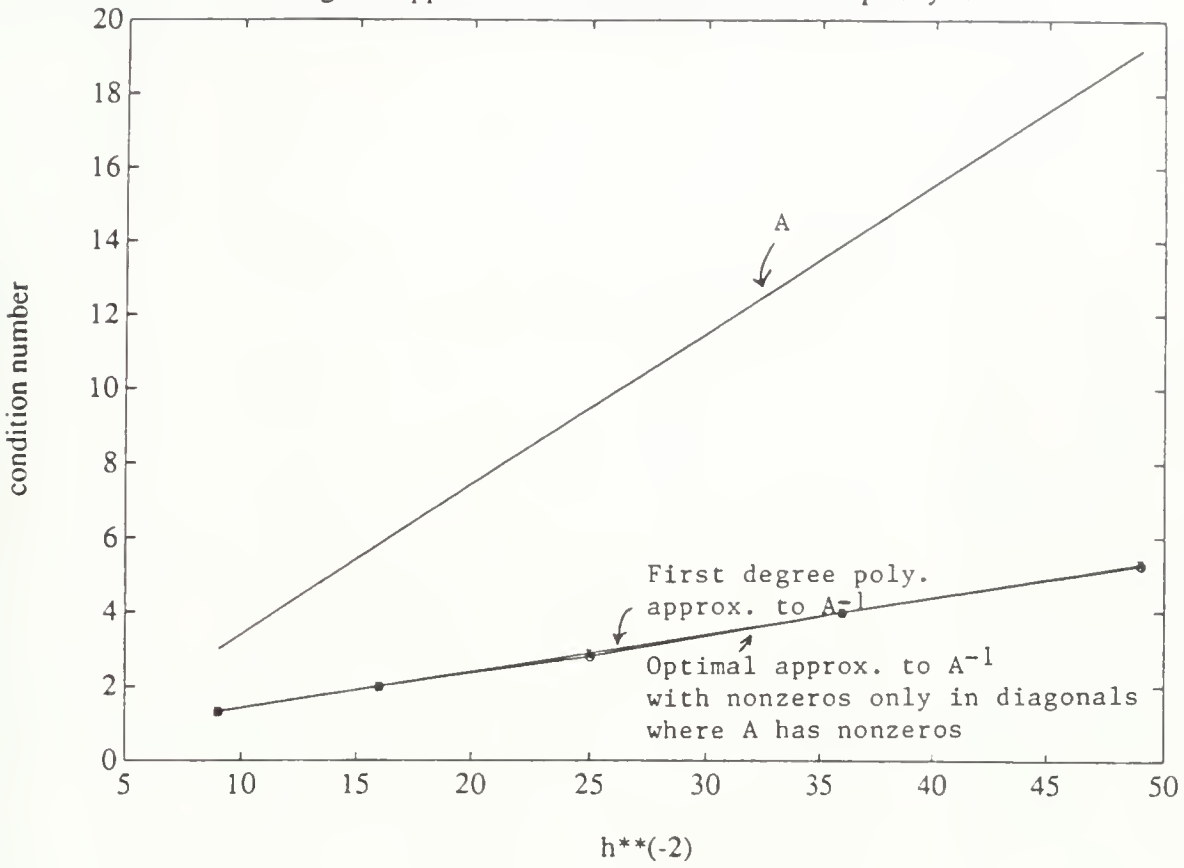


Fig. 3. Tridiagonal Preconditioners for C

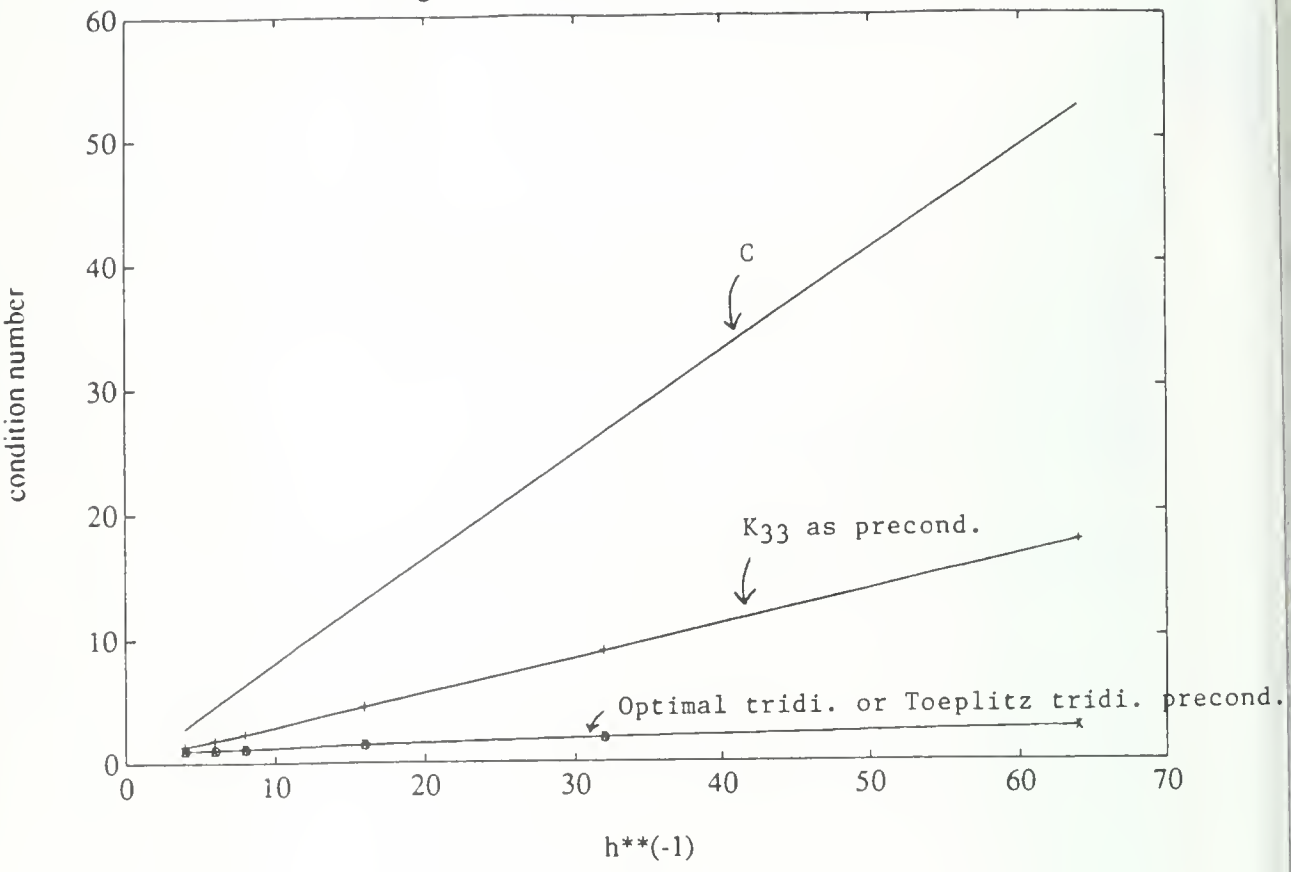


Fig. 3b. Tridiagonal Preconditioners for C

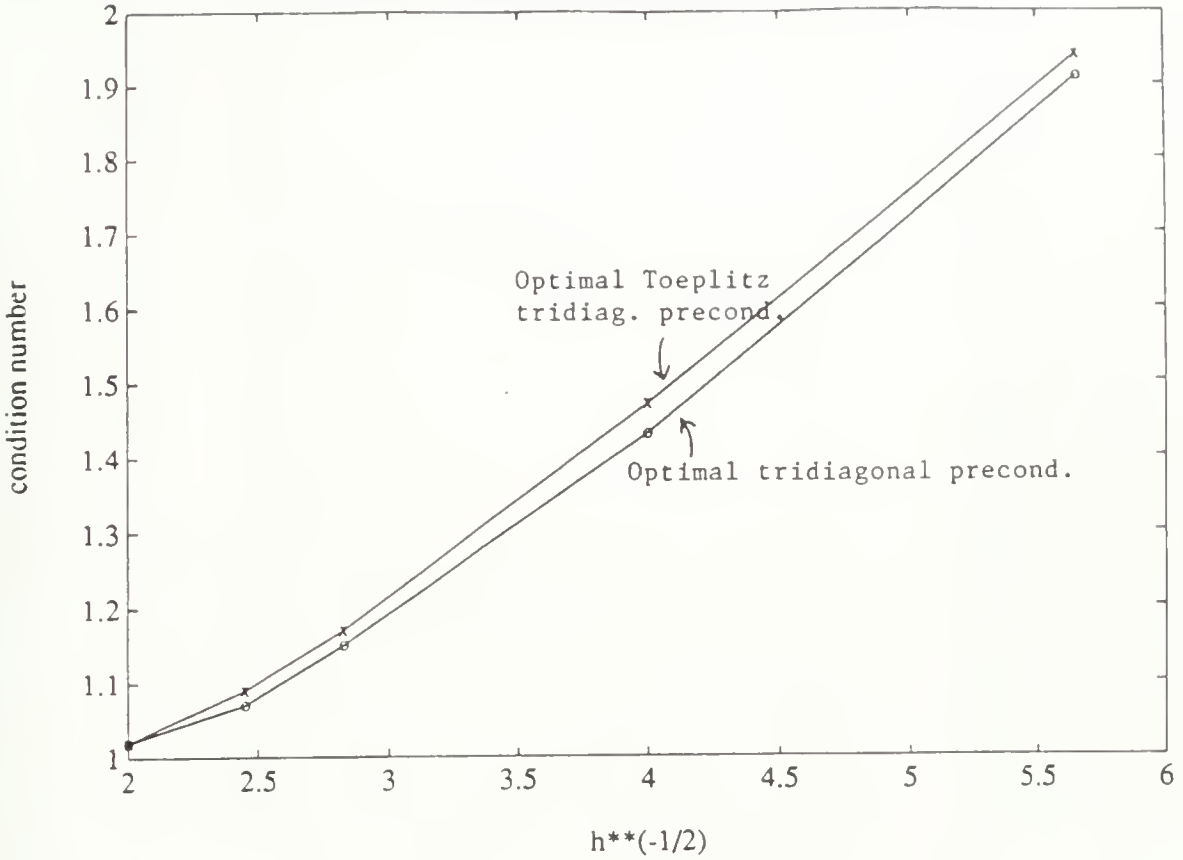


Fig. 4. Toeplitz Preconditioners for C

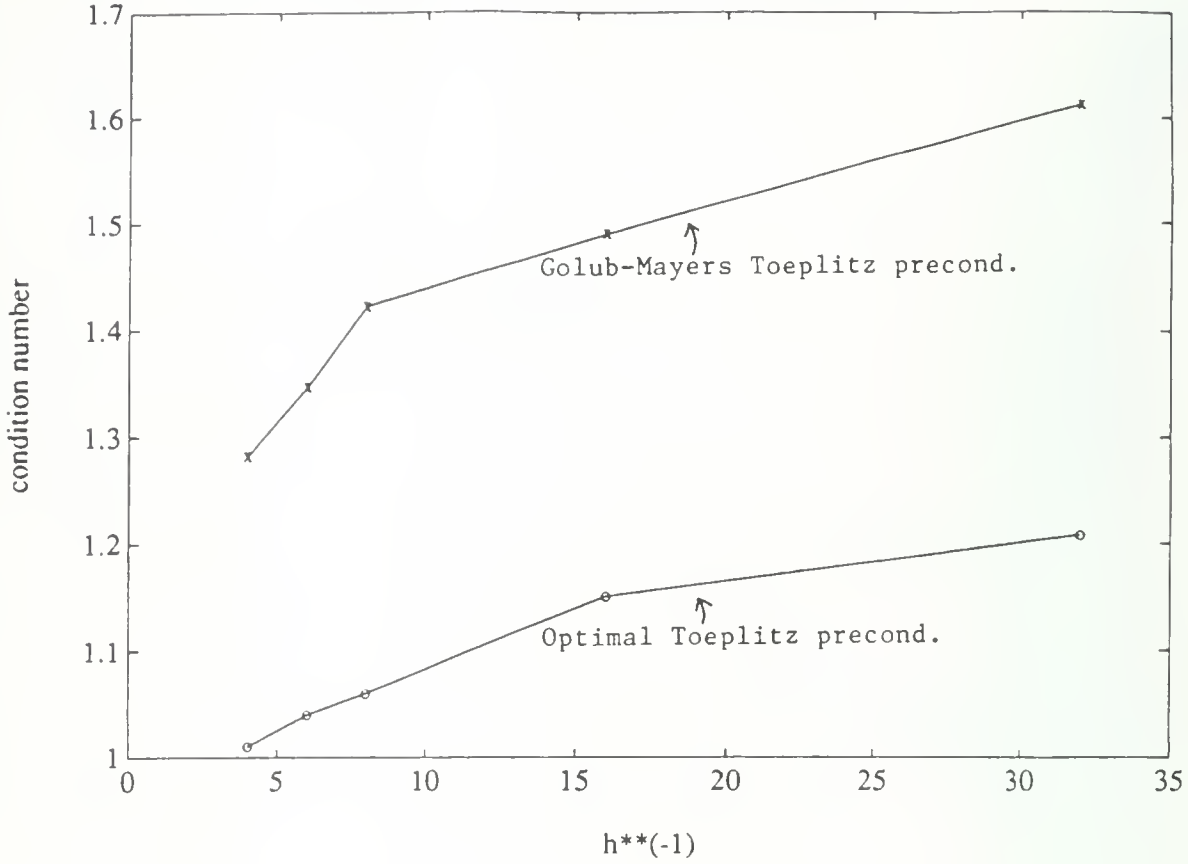


Fig. 4b. Eigenvalues of Golub-Mayers Preconditioner (h=1/16)

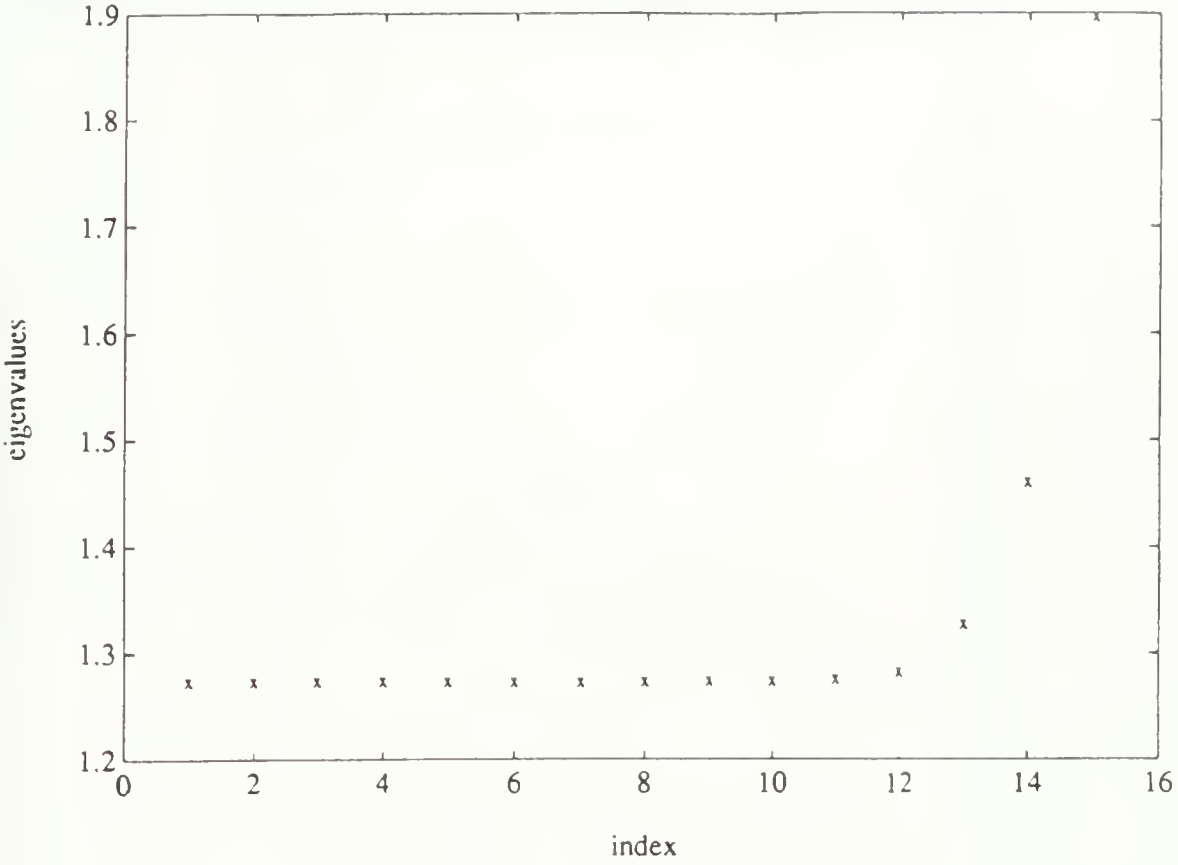


Fig. 4c. Eigenvalues of Optimal Toeplitz Preconditioner (h=1/16)

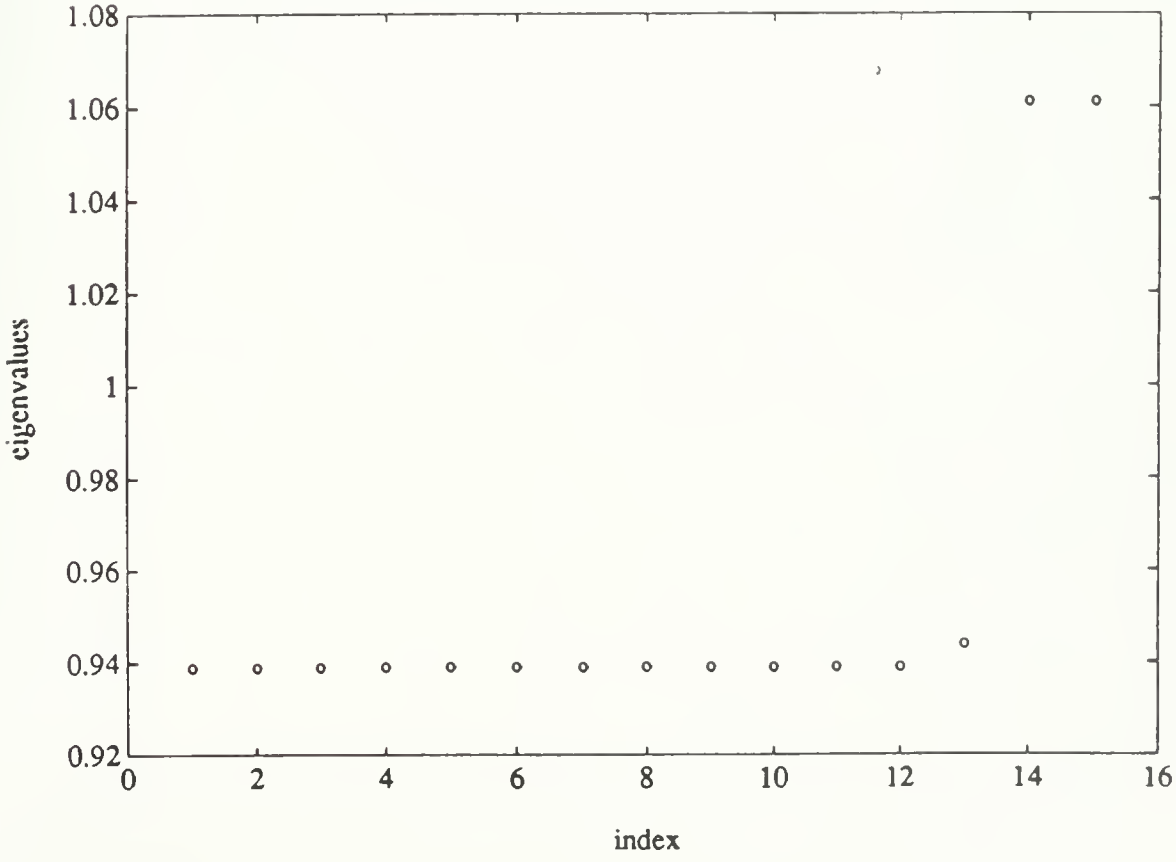


Fig. 5. Preconditioners of the Form KK^T;K Has Sparsity of Lower Tri(A)

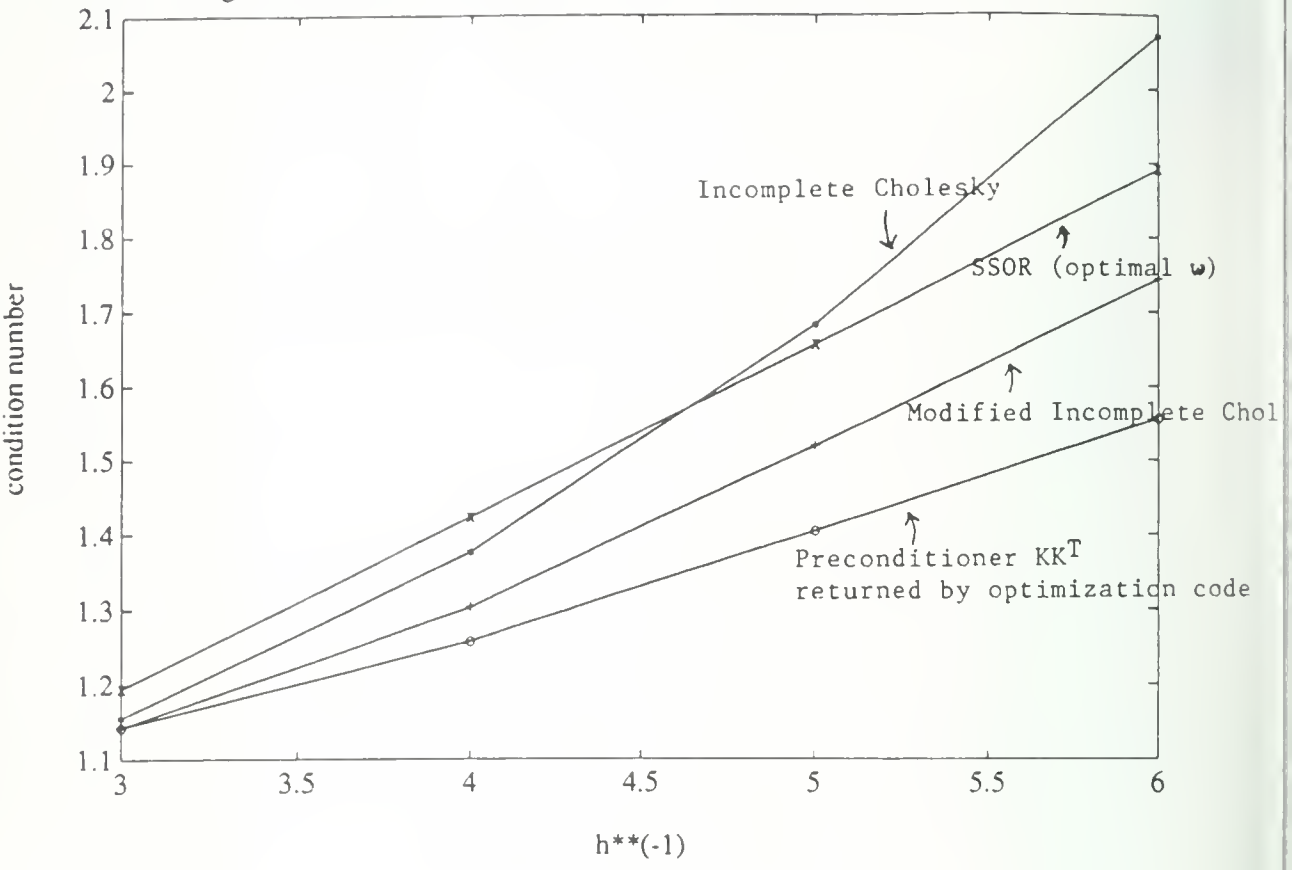


Fig. 6. Diagonally Preconditioned Elliptic Operators

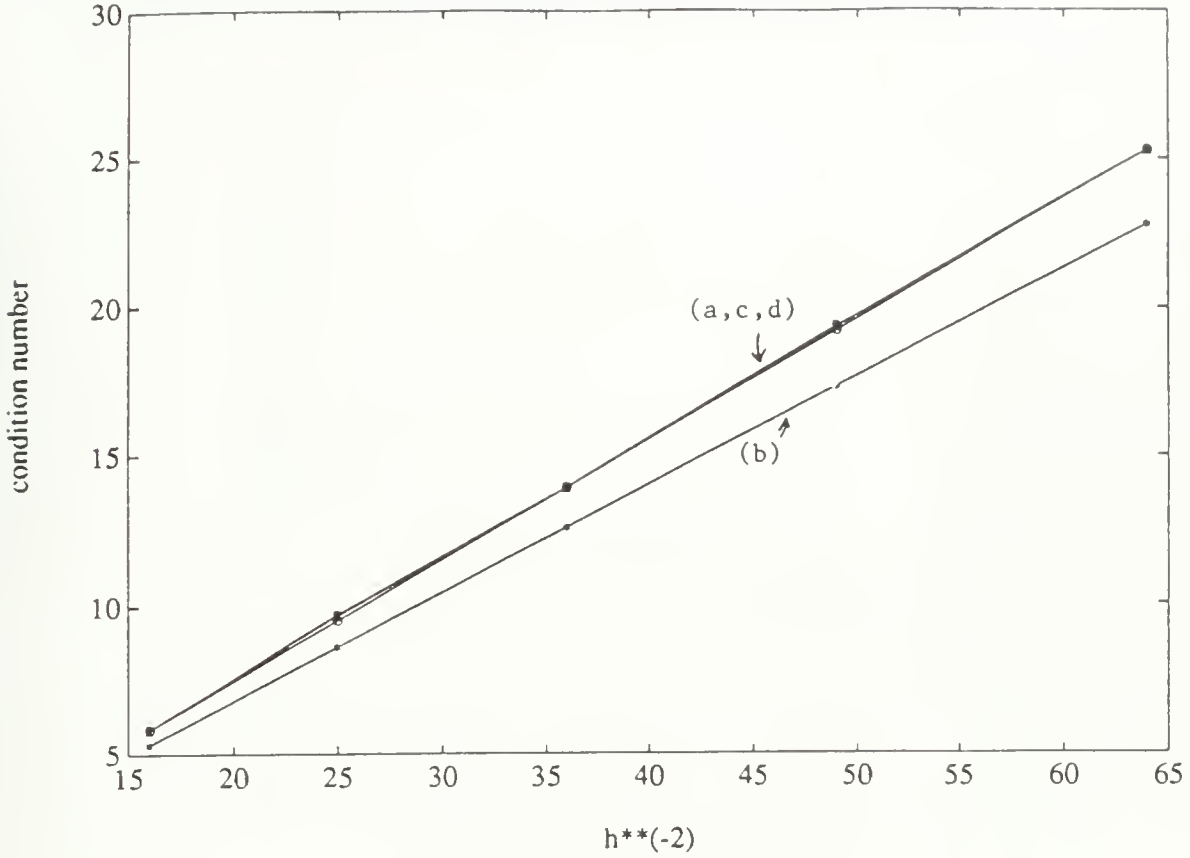


Fig. 7. Tridiagonally Preconditioned Elliptic Operators

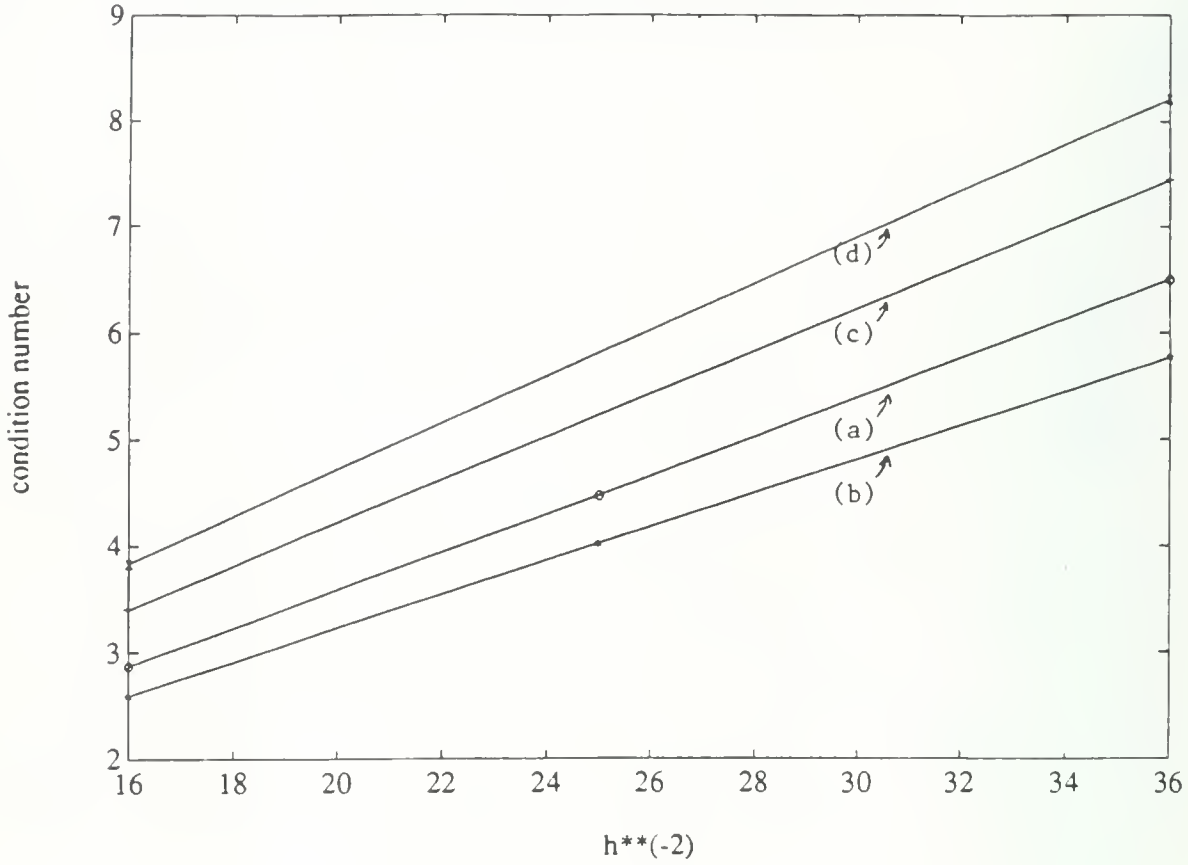
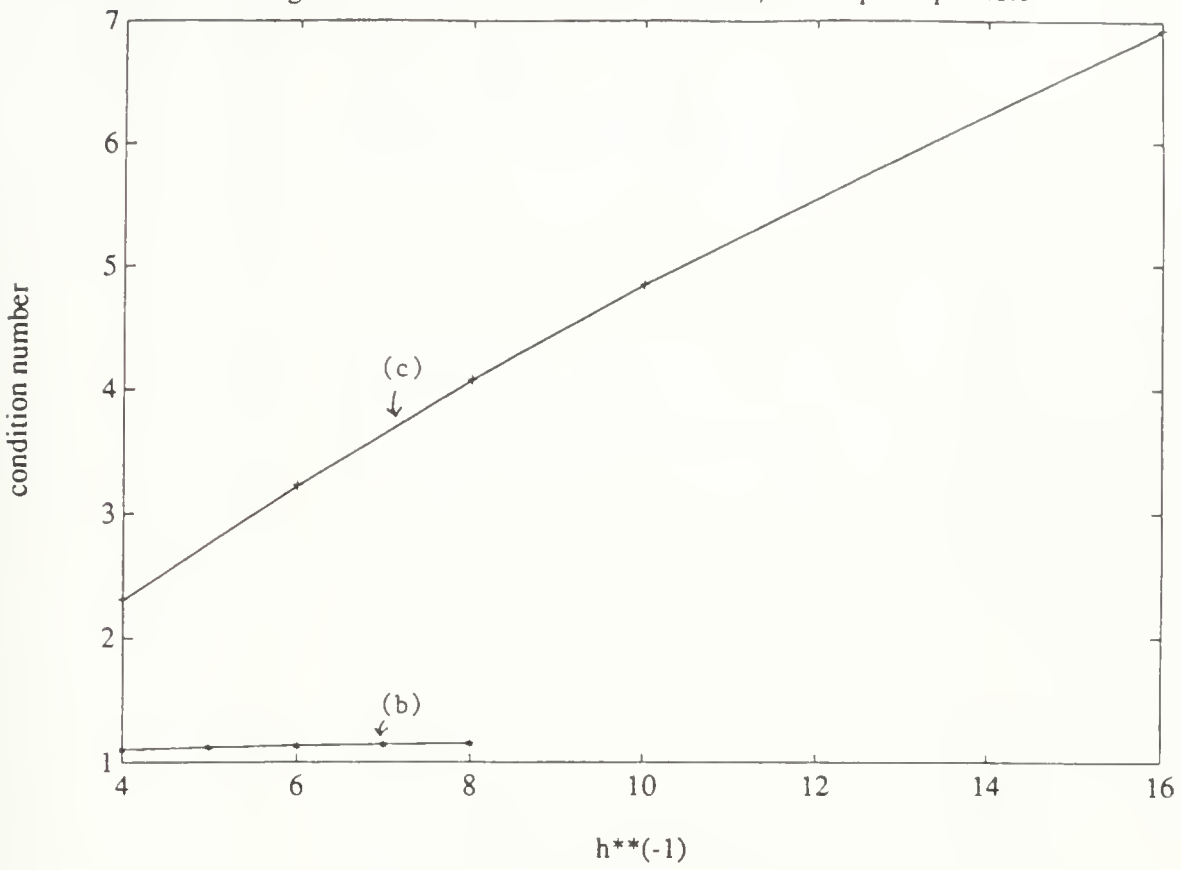


Fig. 8. Preconditioners of the Form DΔD, for Elliptic Operators



NYU COMPSCI TR-431
Greenbaum, Anne
Optimal preconditioners of
a given sparsity pattern
c.2